UNIVERSITÀ DI PISA

Corso di Laurea in Informatica Umanistica

RELAZIONE

Studio della variabilità delle Entità Nominate ed analisi degli errori di un Named Entity Recognizer adattato al dominio della Pubblica Amministrazione

Candidato: *Ludovica Vasile*

Relatore: *Alessandro Lenci*

Correlatore: *Felice Dell'Orletta*

Anno Accademico 2015-2016

Ringraziamenti

A tutti coloro che in un modo o nell'altro mi sono stati accanto, mi hanno aiutata, mi hanno appoggiata o sostenuta durante la realizzazione di questo lavoro, va la mia gratitudine.

Ringrazio innanzitutto i miei genitori, che anche da "lontano" mi sono stati vicino, sostenendomi e offrendomi la possibilità di intraprendere questo percorso.

Ringrazio il Prof. Lenci, mio relatore, per la disponibilità e l'aiuto fornitomi durante la stesura.

Per le medesime ragioni, un caloroso ringraziamento va a Lucia Passaro, che mi ha costantemente e pazientemente seguita con interesse.

Ringrazio infine Maria, che dalla lontana Francia, è venuta qui in questo giorno importante, per me, e Iacopo per avermi sempre sostenuta e incoraggiata e per continuare sempre a farlo.

Indice

1. Introduzione 4

2. Named Entity Recognizer 7

2.1 Stanford NER 8

2.2 Features 11

3. NER in *SEMPLICE* 13

3.1 Progetto *SEMPLICE* 13

3.2 Schema di annotazione 14

3.2.1 IOB format 16

3.2.2 Features in *Semplice* 16

3.2.3 Gazetteers 17

4. Variabilità delle Entità Nominate 19

5. Analisi degli errori 24

6. Analisi computazionali 28

6.1 Indici statistici 28

6.2 Indici di accuratezza di classificazione automatica 29

7. Matrice di Confusione e N-Fold Cross-Validation 30

7.1 Matrice di Confusione 30

7.2 N-Fold Cross-Validation 36

8. Conclusioni 39

9. Bibliografia 40

9.1 Sitografia 43

1. Introduzione

L'oggetto di studio del presente lavoro trova le sue radici in quel settore della *Linguistica Computazionale* (d'ora in poi LC)[1], delineatosi a partire dalla fine degli anni '80, definito *Ingegneria del Linguaggio (Language Engineering)*.

L'ingegneria del linguaggio può essere definita come:

«la disciplina o azione dei sistemi software dell'ingegneria che realizza attività coinvolgendo l'elaborazione del linguaggio umano. Sia il processo di costruzione che il suo output sono misurabili e predicibili.» (Cunningham, 1999)

La crescente quantità di dati ed informazioni reperibili molto spesso sotto forma di testo libero, o in lingue diverse, spesso con alfabeti diversi, quindi in forme tutt'altro che strutturate e capaci di consentire una loro identificazione e catalogazione, nonché il continuo bisogno dell'uomo di misurarsi quotidianamente con un'infinità di testi completamente differenti gli uni dagli altri, ha fatto emergere il bisogno di sviluppare strumenti capaci di *estrarre* le informazioni di cui ha effettivamente bisogno. Da qui la nascita ed il continuo progredire del settore dell'ingegneria del linguaggio.

Un elemento chiave di questo settore sono i processi di *Information Extraction* (o IE), basati sul *Natural Language Processing* (o NLP).

Come suggerisce lo stesso nome, il NLP si riferisce all'elaborazione informatica del linguaggio naturale, indipendentemente dallo scopo o dal livello di approfondimento dell'analisi. Si tratta di sistemi in grado di accedere al contenuto di informazione, dotando il computer di conoscenze complesse sulla struttura del linguaggio naturale, permettendo l'estrazione di informazioni dai testi, o il potenziamento della capacità di ricercare documenti rilevanti per l'utente. Il concetto di NLP è strettamente correlato a quello di LC. NLP e LC costituiscono in un certo senso due facce della stessa medaglia: mentre il focus del NLP sono le applicazioni, quello della LC è lo studio del linguaggio con metodologie quantitative e informatiche.

[1] La *Linguistica Computazionale* è una disciplina che si concentra sullo studio e sull'elaborazione del linguaggio naturale, in particolare sullo sviluppo di formalismi descrittivi del funzionamento del linguaggio naturale, tali che possano trasformarsi in programmi eseguibili dai computer.

Un altro concetto spesso correlato a NLP è quello di *Apprendimento Automatico* (d'ora in poi ML, da *Machine Learning*), definito come:

«il campo di studio che dà ai computer l'abilità di apprendere senza essere esplicitamente programmati a farlo.» (Samuel, 1950)

Tom M. Mitchell, più di recente, ha fornito una definizione formale del termine:

«un programma apprende da una certa esperienza E se: nel rispetto di una classe di compiti T, con una misura di prestazione P, la prestazione P misurata nello svolgere il compito T è migliorata dall'esperienza E.»

In realtà il ML, naturalmente va ben oltre l'ambito del NLP: gli algoritmi di apprendimento automatico utilizzati per diversi casi di elaborazione del linguaggio possono essere ugualmente utilizzati per risolvere altri problemi di *Intelligenza Artificiale*[2]. Nel caso del ML più propriamente, si forniscono al computer algoritmi che gli permettono di svolgere un compito a partire da un campione di esempi rappresentativi (corpus di addestramento): spetta al sistema ricavare le regole per svolgere il compito.

Un'altra delle sub-discipline dell'ingegneria del linguaggio che ultimamente sta prendendo piede è, come accennato sopra, l'Information Extraction.

L'IE, rappresenta la più frequente applicazione di NLP e, dotato di conoscenze non banali sulla struttura sintattica, sia sull'aspetto semantico delle frasi, cerca di applicare metodi e tecnologie informatiche per l'estrazione di nuclei di informazione strutturata o semi strutturata, a partire da testi non strutturati in formato digitale.

Altro concetto, strettamente legato a quello di IE è quello di *Named Entity Recognizer* (o NER), che può essere definito come una fase dell'IE.

Tale premessa risulta fondamentale per comprendere quello che è l'oggetto di studio del presente lavoro: *lo studio della variabilità delle entità nominate*[3]*, e l'analisi degli errori di un Named Entity Recognizer adattato al dominio della Pubblica Amministrazione.*

[2] *L'Intelligenza Artificiale* (o AI da *Artificial Intelligence*) è una disciplina recente, che negli anni ha fornito un importante contributo al progresso dell'intera informatica e che può essere definita come l'abilità di un computer di svolgere funzioni o ragionamenti tipici della mente umana.

[3] Una *entità nominata* (o NE da *named entity*) è una frase, un'espressione che identifica un elemento da un insieme di altri elementi che hanno "caratteristiche" simili. Esempi di entità nominate sono nomi e cognomi, luoghi geografici, età, indirizzi, numeri di telefono, le aziende o organizzazioni.

Il lavoro parte dal processo di addestramento dell'algoritmo di apprendimento automatico su un corpus (il *training set*[4]) appartenente al dominio amministrativo, dapprima annotato manualmente, sulla base di uno schema di annotazione predefinito.

Per quanto riguarda lo studio della variabilità delle entità nominate, per ogni classe semantica sono state analizzate le varie forme attraverso le quali ciascuna entità nominata può presentarsi.

Per quanto riguarda l'analisi degli errori, questa è stata realizzata su un nuovo corpus (il *test set*), costituito di 25 documenti, previamente annotati sulla base dello stesso schema di annotazione utilizzato per il corpus di training originale. In un primo momento sono stati presi in considerazione gli output risultanti dall'applicazione dell'algoritmo di riconoscimento automatico, per un vero e proprio confronto tra gli errori commessi dalla macchina (quindi dal NER) e gli errori commessi durante l'annotazione manuale.

I risultati dell'addestramento sono stati analizzati a livello computazionale attraverso il calcolo della *Matrice di Confusione* e l'analisi degli indici di accuratezza di classificazione statistica.

Attraverso l'applicazione della tecnica statistica della *Cross-Validation* al corpus di training, è stato possibile valutare l'affidabilità del modello.

Altri indici statistici relativi alla lunghezza del corpus in token, al numero di *parole contenuto* (o *content words* in inglese)[5] ed al numero di entità nominate presenti nel corpus, sono stati calcolati, come stime prettamente *quantitative* relative al training set.

Per le analisi computazionali, sia per il calcolo degli indici statistici, sia per il calcolo degli indici di accuratezza di classificazione statistica, il linguaggio di programmazione scelto è stato *Python*[6].

[4] vedi cap. 3, par. 3.1.
[5] All'interno del repertorio lessicale si distinguono due classi principali: le *parole contenuto* e le *parole funzione* (o *funtori*). Mentre quest'ultime (identificabili con articoli, pronomi, congiunzioni e preposizioni) svolgono funzioni supplementari (per esempio, indicare le relazioni esistenti tra le parole che introducono il contenuto), le prime forniscono il contenuto vero e proprio. A questa classe appartengono nomi (S), verbi (V), aggettivi (A) ed avverbi (B).
[6] Python è un linguaggio di programmazione ad alto livello, orientato agli oggetti ed interpretato.

2. Named Entity Recognizer

Il *Named Entity Recognizer* rappresenta, nell'ambito del NLP, un importante sub-task dell'IE, attraverso cui informazione strutturata viene estratta da testo non strutturato, come articoli o documenti vari.

Il termine *named entity*, ora ampiamente adoperato nell'ambito del NLP è stato coniato per la *Sixth Message Understanding Conference* (MUC-6)[7] (R. Grishman, B. Sundheim, 1996). A quel tempo, MUC è stato incentrato propriamente sui compiti dell'IE, compiti in cui informazioni strutturate delle attività aziendali o relative alla difesa, vengono estratte da testi non strutturati. In occasione della sesta conferenza (MUC-6) venne definito il task relativo al riconoscimento delle entità nominate. Nel definire il task è stato visto che è essenziale riconoscere unità informative, come nomi, incluse persone, organizzazioni, nomi di luoghi, espressioni numeriche compreso il tempo, la data, o la quantità. Indentificare i riferimenti a queste entità è stato riconosciuto come uno dei più importanti sub-task dell'IE ed è stato più propriamente definito «Named Entity Recognition and Classification (NERC)» (David Nadeau, Satoshi Sekine, 2007).

La ricerca computazionale, volta ad individuare automaticamente le entità nominate nei testi, dispone di un gran numero di strategie, metodi e rappresentazioni. Il primo lavoro di ricerca nel campo, è stato presentato alla *Seventh IEEE Conference on Artificial Intelligence Applications* da Lisa F. Rau (1991): nell'articolo viene descritto un sistema per *estrarre e riconoscere* nomi di aziende e si basa su euristiche e regole realizzate manualmente.

Dal 1991 al 1995 il tasso di pubblicazione è rimasto relativamente basso per poi accelerare nel 1996 con il primo grande evento dedicato al task: il MUC-6. Da allora non è mai diminuito, con costanti ricerche e numerosi eventi scientifici.

Il compito del NER può essere così scomposto in due sotto-problemi, il primo riguardante l'*identificazione* dei nomi propri, e il secondo circa la *categorizzazione*

[7] Le *Message Understanding* Conferences vennero avviate e finanziate da DARPA (Defense Advanced Research Project Agency) al fine di promuovere lo sviluppo di nuovi e migliori tecniche di IE. La competizione si articolava nello sviluppo di standard per la valutazione, come ad esempio l'adozione delle metriche di *precisione* e *richiamo*.

semantica vera e propria. Dopo l'estrazione, quindi l'identificazione dei nomi propri che compaiono nel testo, il NER procede a classificare tali entità in una serie di categorie (definite anche *classi*) semantiche di interesse definite a priori. Nel termine *named entity*, la parola *named* mira dunque a limitare il compito a quelle entità che sono descritte da uno o più «designatori rigidi» (Kripke, 1982)[8], cioè i nomi propri. Le tre categorie universalmente riconosciute nella classificazione delle NE riguardano le persone (PER), i luoghi (LOC) e le organizzazioni (ORG).

Le applicazioni di questo strumento nell'ambito del NLP sono molteplici, come la traduzione automatica, i sistemi di *question answering*[9], l'indicizzazione e il recupero delle informazioni, la classificazione dei dati e la *text summarization*[10] (Passaro, 2014).

Per il NER sono stati proposti diversi approcci: sistemi *Rule-Based*, sistemi *Machine-Learning* (tra cui *Hidden Markov*[11] *Model*, *Maximum Entropy Model*, *Decision Trees*, *Support Vector Machines* e *Conditional Random Field*) ed approcci ibridi.

A differenza dei sistemi basati su regole che in linea di massima danno buoni risultati, ma richiedono un elevato tempo di sviluppo da parte di esperti linguisti, le tecniche di apprendimento automatico utilizzano invece una raccolta di documenti annotati per addestrare i classificatori, e quindi il tempo di sviluppo si sposta dalla definizione manuale di regole, verso la preparazione di corpora annotati.

2.1 Stanford NER

Lo *Stanford NER* è una implementazione Java di NER e disponibile per il download, sotto licenza GNU General Public License[12]. Inclusi nel download vi sono ottimi NER

[8] Una delle tesi fondamentali di Kripke è che i nomi sono "designatori rigidi". Un designatore rigido è un termine che si riferisce alla medesima entità in tutti i mondi possibili. Ciò vuol dire che, una volta fissato il referente di un nome, questo, anche cambiando la situazione (si parla in questo senso di "mondi possibili"), non cambia più.
[9] Nell'ambito dell'*Information Retrieval*, i sistemi di *question answering* si pongono l'obiettivo di trovare la risposta ad una domanda posta in linguaggio naturale, partendo da una raccolta di documenti (un corpus, o un'altra collezione di testi).
[10] La *text summarization* è definita come il processo di *sintetizzazione* di informazione più importante, a partire da una sorgente, per produrre una versione ridotta per un particolare utente o task.
[11] I *Modelli di Markov* permettono di creare modelli probabilistici di sequenze linguistiche per le quali si assume che esiste un particolare tipo di dipendenza tra gli elementi della sequenza stessa. In un modello di Markov di ordine n, la probabilità che venga prodotta una certa parola è calcolata come il prodotto della probabilità di n+1 – grammi. Dipende, in altre parole, solo da un numero limitato di parole precedenti.
[12] La GNU General Public License è una licenza copyleft per il software libero, nata originariamente per patrocinare i programmi creati per il sistema operativo GNU. Un programma protetto da GNU GPL, anche col susseguirsi di modifiche, deve rimanere libero.

per la lingua inglese, in particolare per 3 classi (persone, organizzazioni e luoghi). Il software è stato sviluppato da Jenny Finkel presso l'Università di Stanford. Le funzioni di estrazione sono ad opera di Dan Klein, Christopher Manning, e Jenny Finkel.

Lo Stanford NER, conosciuto anche come *CRFClassifier*, si è evoluto nel corso degli anni ed è basato appunto su un *Conditional Random Field* (CRF) (Lafferty et al., 2001). I modelli CRF rappresentano una classe di metodi di modellazione statistica spesso applicati a schemi di riconoscimento ed apprendimento automatico, essendo utilizzati per la previsione strutturata, per l'etichettatura o l'analisi di dati sequenziali. Mentre un ordinario classificatore prevede un'etichetta per ogni singolo campione a prescindere dai campioni *vicini*, un CRF può prendere in considerazione il contesto intero, consentendo di generare un flusso di probabilità per tutta un'intera sequenza.

Nell'ambito del *sequence modeling* il grafico di interesse è solitamente un grafo a catena: una sequenza in input di variabili osservate X, rappresenta una sequenza di osservazioni e Y rappresenta una sequenza di variabili sconosciute (o stati nascosti), da dedurre viste le osservazioni. Le variabili Y_i sono strutturate a formare una catena con degli archi tra Y_{i-1} e Y_i. In altre parole, si tratta di un metodo volto a modellare la *probabilità condizionata*[13] di una sequenza di stati nascosti, data una sequenza di osservazioni.

Una disposizione di questo tipo ammette, per l'interpretazione della sequenza Y, una serie di algoritmi efficienti: algoritmi per l'apprendimento (determinare la distribuzione di probabilità di transizione tra le Yi, e le *features*[14] che spiegano tale distribuzione), di inferenza (determinare la probabilità di una *label*[15] per Y, dato X) e di decodifica (determinare la label più probabile per Y, dato X).

Gli *hidden state sequence models*, come ad esempio *Hidden Markov Model* (HMM)[16], *Conditional Markov Model* (CMM)[17], e *Conditional Random Fields* (CRF) rappresentano un importante approccio per il compito dell'IE. Tali modelli sfruttano

[13] La *probabilità condizionata* di un evento A rispetto un evento B, definita come P(A|B), rappresenta la probabilità che si verifichi A, sapendo che è avvenuto B.
[14] v. par. 2.3.
[15] Una l*abel* è un'etichetta che, nel caso della classificazione, coincide con la classe di assegnazione di Y.
[16] Un HMM è una catena di Markov in cui ogni stato, non osservabile direttamente, genera un evento secondo una certa distribuzione di probabilità che dipende solo dallo stato.
[17] Un CMM (detto anche *maximum-entropy Markov model*) è un modello che combina le caratteristiche di HMM e di *Maximum Entropy Model.* Estende un *Maximum Entropy classifier* (un classificatore utilizzato tipicamente in problemi di NLP o IR), ipotizzando che le incognite da conoscere sono connesse attraverso una catena di Markov piuttosto che essere condizionalmente indipendenti tra loro.

la proprietà markoviana secondo la quale, decidere su uno stato in una particolare posizione della sequenza, può dipendere solo da una piccola finestra di altri stati intorno ad esso.

Nell'ambito dell'IE, talvolta è utile modellare anche delle caratteristiche non-locali, ossia non legate ad una finestra di contesto prestabilita. Per esempio, osservando la frase:

"*Il giornale Repubblica ha riportato la notizia"*

risulta semplice utilizzare come *feature* la presenza di *"il giornale"* per classificare correttamente *"Repubblica"* come ORG. Osservando poi una seconda frase che compare nel testo:

"*Repubblica ha scritto a proposito di…"*

pur essendoci un riferimento meno esplicito rispetto all'esempio precedente, l'informazione appresa può essere utilizzata per classificare correttamente *"Repubblica"* come ORG.

Per l'inferenza della sequenza di stati nascosti più probabili dato l'input, sono possibili approcci diversi. Uno standard è rappresentato dall'*algoritmo di Viterbi*[18]. L'algoritmo di Viterbi è un algoritmo che, basandosi sulla *proprietà di Markov* secondo la quale la probabilità di trovarsi, in un determinato istante, in uno stato, dipende solo dallo stato all'istante immediatamente precedente, viene comunemente impiegato per trovare la migliore sequenza di stati nascosti, in una sequenza di eventi.

Lo Stanford NER propone invece l'uso del *Gibbs Sampling* (o *Campionamento di Gibbs*), un algoritmo statistico, per eseguire l'inferenza tenendo conto della struttura non locale e mantenendo gestibile la complessità computazionale. In altre parole, per ottenere una sequenza di campioni casuali a partire dalla distribuzione di *probabilità congiunta*[19] di due o più variabili casuali. Il Gibbs Sampling fa parte dei *metodi Monte Carlo* (o *Markov chain Monte Carlo*).

[18] Per approfondimenti, si consulti Viterbi A.J. (1967), Rabiner (1989) .
[19] La *probabilità congiunta* di due eventi A e B, definita anche P(A∩B), è pari alla probabilità di uno dei due eventi, moltiplicata per la probabilità condizionata dell'altro rispetto al primo.

I metodi Monte Carlo sono una classe di metodi computazionali (algoritmi per il campionamento) basati sul campionamento casuale, per l'inferenza approssimata. Immaginando di avere un modello M a stati nascosti che definisce una distribuzione di probabilità sulle sequenze di stati e condizionato da un dato input, è possibile, data una sequenza osservata di input o = {o0,. . . , On}, calcolare la probabilità condizionata PM (s | o) di una qualsiasi sequenza stati s = {s0, … , Sn}.

Data la natura stocastica del Gibbs Sampling, i risultati prodotti possono essere differenti ogni volta che l'algoritmo viene eseguito. Esso rappresenta inoltre un'alternativa agli algoritmi deterministici utilizzati nell'inferenza statistica.

2.2 Features

Gli attributi, le caratteristiche delle parole utili ai fini dell'apprendimento automatico, prendono il nome di *feature*. Per rappresentare una parola attraverso un insieme di features, possono essere impiegati valori booleani, numerici e nominali.

Per esempio, in un NER possono essere usate features del tipo:

- attributo booleano settato a *true* se la parola è scritta con la lettera maiuscola, *false* altrimenti
- un attributo numerico corrispondente alla lunghezza in caratteri della parola
- un attributo nominale corrispondente alla parola minuscola, o al lemma.

Tre macro-gruppi identificano le features interessanti per il riconoscimento di NE.

1) Features a livello di parola: descrivono tratti ortografici come la prima lettera maiuscola, la punteggiatura per gli acronimi, caratteri speciali o numerici. Per esempio, *"s.r.l."* o *"s.p.a."* tra le parole che identificano una entità di tipo ORG sono molto utili alla determinazione della classe.

2) *Gazetteers*: sono features rappresentate attraverso liste di parole associate a una classe. In molti sistemi statistici questi vengono usati solo come "indizio", dal momento che spesso gli stessi nomi presenti nelle liste non possono essere classificati prescindendo dal contesto.

   Per esempio, *"San Giovanni Battista"* può essere un nome di via o un nome di persona:

   "La storia di [San Giovanni Battista] **PER** *comincia negli anni…"*

   "L'immobile si trova in via [San Giovanni Battista] **LOC**"

3) Features a livello di documento: queste riguardano altre entità nel contesto, la posizione della NE nel documento, i metadati e così via.

3 NER in *SEMPLICE*

3.1 Progetto *SEMPLICE*

Il progetto *SEMPLICE* (*Semantic Web for Public Administrators and Citizens*) nasce nel 2012, grazie al finanziamento ottenuto dalla Regione Toscana per la realizzazione di un'idea imprenditoriale avanzata da un gruppo di aziende ed enti di ricerca e capeggiato dalla Community Company 01S S.r.l., in collaborazione con l'Università di Pisa – Dipartimento di Filologia, Letteratura e Linguistica, che ha contribuito attivamente alla realizzazione degli strumenti di analisi semantica. Il progetto è attualmente un prototipo in fase di industrializzazione: la start up Eti3 è nata da 01S specificatamente per la gestione delle attività legate al progetto *Semplice*, con il supporto e la collaborazione dell'Università di Pisa.

Per l'addestramento dello Stanford NER è stato adoperato un *training set*[20] costituito di 460 documenti appartenenti al dominio della Pubblica Amministrazione. Attraverso degli strumenti di *crawling* (analisi dei contenuti), i documenti sono stati reperiti dagli albi pretori dei Comuni italiani: la tipologia di questi atti è molto variegata, ma si tratta perlopiù di Determinazioni, Ordinanze, Decreti e Delibere. I documenti sono stati inoltre selezionati in maniera casuale, ma rappresentativa allo stesso tempo: per ogni Comune sono stati scelti circa 3 o 4 documenti.

Il test è stato eseguito su un nuovo corpus, il *test set*, costituito di 25 documenti, appartenenti a 25 entità amministrative differenti, ed annotati sulla base dello stesso schema di codifica utilizzato per l'annotazione del training set.

[20] Nell'ambito del ML, l'*apprendimento supervisionato* rappresenta una tecnica che consiste nell'istruire un sistema informatico in modo da consentirgli di risolvere dei compiti in maniera autonoma, sulla base di una serie di esempi (un insieme di dati pre-etichettati) che gli vengono forniti inizialmente e che vanno a costituire il *training set*.

3.2 Schema di annotazione

Per la classificazione delle entità nominate, è stato impiegato uno schema di annotazione articolato in 6 classi semantiche, e cioè:

1) Person (PER)
2) Organization (ORG),
3) Location (LOC),
4) Law (LAW),
5) Act (ACT),
6) Organization PA (ORG_PA).

Le classi semantiche possono essere descritte come segue.

Person (PER): l'entità può riferirsi ad un singolo individuo o ad un gruppo di individui. L'entità viene annotata anche nel caso in cui viene riscontrata nel testo come acronimo, nickname o nome di famiglia. Esempi di annotazione di questo tipo di entità possono essere:

*"F.to il responsabile [Andrea Fracassi]***PER**"

*"Come richiesto dai Sigg. [Aversano]***PER**"

*"Papa [Benedetto XVI]***PER**"

Organization (ORG): le organizzazioni rappresentano un mix molto eterogeneo, aziende, ditte, scuole, musei e altri gruppi di persone definiti da una struttura organizzativa consolidata. Esempi di queste entità sono organizzazioni governative e ministeri, organizzazioni giudiziarie, organizzazioni commerciali, organizzazioni di intrattenimento, organizzazioni non governative, compagnie assicurative ecc.

*"L'agenzia [Unipol Assicurazioni]***ORG** *opera attraverso cinque divisioni"*

*"Secondo i dati raccolti dall'[ISTAT]***ORG**"

*"L'[Amministrazione Comunale]***ORG** *ha così deliberato"*

Location (LOC): rientrano in questa classe sia le entità geografiche come aree geografiche, indirizzi, località, sia le entità geopolitiche come nazioni, regioni, province, comuni. Esempi di LOC sono:

*L'immobile è sito in [Via Nazionale 12]***LOC**

*[Provincia di Torino]***LOC**

*Lo scenario ligure delle[Cinque Terre]***LOC**

Law (LAW): tutti i riferimenti che presentano rango normativo, quindi leggi dello stato (come decreti leggi, decreti legislativi, decreti del presidente della repubblica, testi, unici, codici, ecc.), decreti ministeriali, leggi regionali. Esempi di LAW sono:

*Ai sensi dell'[art. 2 del Decreto Legge 65/02]***LAW**

*Così come emanato dal [Codice della Strada]***LAW**

*Richiamato il [Decreto del Ministero della Salute]***LAW**

Act (ACT): appartengono alla classe degli ACT solo gli atti del Comune e in particolare delibere, determine, ordinanze, decreti. Tutti gli atti emanati da autorità diverse, come gli atti della Regione o delibere di autorità varie, non vengono annotati. Inoltre a seconda della "funzione semantica" svolta, per ogni parola che costituisce l'entità ACT, vengono impiegati tag diversi: ACT_T (Act Type), ACT_X (parole aventi funzione puramente sintattica come "num.", "del" e segni di punteggiatura), ACT_N (Act Number), ACT_D (Act Date) e ACT_U (nel caso di elementi costituiti da più tag, come per esempio numero e data in un'unica espressione). Esempi di ACT sono:

*Vista la [Delibera del Responsabile]***ACT_T** *[num.]***ACT_X** *[65]***ACT_N** *[del]***ACT_X** *[02-12-12]***ACT_D**

*Approvata con [Delibera di Giunta]***ACT_T** *[65/08]***ACT_U**

*[Ordinanza comunale]***ACT_T** *[n.]***ACT_X** *[141/15]***ACT_U**

Organization PA (ORG_PA): insieme di partizioni organizzative (ufficio, settore, direzione, servizio, area, ecc.) in cui è articolato il Comune. Esempi di ORG_PA sono:

*F.to il responsabile del [Servizio Finanziario]***ORG_PA**

*I [Servizi Sociali]***ORG_PA** *del Comune di Scicli*

*[Organismo Indipendente di Valutazione]***ORG_PA**

3.2.1 IOB format

Il formato IOB (*Inside Outside Beginning*) è un formato di codifica comunemente utilizzato per l'annotazione di tokens nell'ambito della LC (e in particolare del NER).

Il prefisso B (da *beginning*) prima del tag indica che il tag costituisce l'inizio del *pezzo* da annotare; il prefisso I (da *internal*) indica invece che il tag è collocato all'interno del pezzo da annotare.

3.2.2 Features in *Semplice*

Al fine di predire nella maniera più efficace la classe corretta di una NE, l'addestramento del modello è avvenuto a partire da una serie di features di diversa natura. Le features che sono state utilizzate (Passaro, 2014) nell'ambito del progetto riguardano:

1. Features di parola

   Parola seguente, parola precedente o una finestra di parole. Per esempio, nella frase

   *Il sindaco [Giorgio Cantoni]***PER** *ha deliberato quanto segue.*

   la presenza della parola "*sindaco*" aiuta nel determinare le parole seguenti come appartenenti alla classe PER.

Allo stesso modo, nella frase

*La fattura è stata emessa dalla ditta [System Data s.r.l.]***ORG**

il suffisso "*s.r.l."* è utile per stabilire che il nome è un'organizzazione.

2. Features ortografiche

   La presenza dei tratti ortografici (ad esempio la lettera maiuscola), aiuta nel determinare che si tratti di una NE.

   *E' stato detto da [Felice Antonioli]***PER**

3. Features linguistiche

   Il NER è stato addestrato fornendo in input informazioni linguistiche derivanti dai moduli di analisi linguistica di *sentence splitting*, *lemmatizzazione*[21], *POS-tagging*[22]. In particolare:

   - Posizione della parola all'interno della frase (attributo numerico)
   - Lemma della parola (attributo nominale)
   - *Part of speech* della parola (attributo nominale)

3.2.3 Gazetteers

Un gazetteer è costituito da un set di liste contenenti nomi di entità, come nomi di persona, di organizzazioni, di luoghi, di entità geopolitiche. Liste di questo tipo

[21] La lemmatizzazione viene definita come il processo di riduzione di una forma flessa di una parola, al lemma.
[22] Il *POS-tagging* (da *Part of speech tagging*) consiste nell'annotazione morfosintattica della parola.

vengono utilizzate nell'ambito del NER tipicamente per rintracciare le occorrenze di tali entità nei testi.

Nella presente versione dell'esperimento non è stato utilizzato alcun gazettier. Per una versione precedente, per esempio, sono stati estratti i nomi di tutti i comuni italiani e inseriti nella lista relativa alle GPE (*geo-political entity*).

4. Variabilità delle Entità Nominate

Per ogni classe semantica, le entità nominate ad essa appartenenti possono presentare forme e strutture differenti. Utile ai fini dell'annotazione, è lo studio della variabilità delle forme attraverso le quali le entità nominate si presentano nel corpus. Di seguito si riportano, per ogni classe semantica, alcuni casi e rispettivi esempi.

Person (PER): i nomi di persona vengono annotati sia quando ad essere espresso è solo il nome, sia quando è solo il cognome, sia quando sono entrambi, nel caso in cui al nome abbreviato (iniziale puntata) segue il cognome, sia nel caso in cui un titolo si trova interposto tra nome e cognome, sia infine nel caso in cui a causa di un errore ortografico il titolo è adiacente al cognome o al nome. Accade spesso che una determinata persona viene citata in modi diversi.

*Come espressamente chiesto dalla signora [Sara]***PER**

*Approvato dal Presidente [G. Galliano]***PER**

*Considerato quanto stabilito dal [Dott.Mambrini Carlo]***PER**

Assiste alla seduta il Segretario Comunale dott.ssa [C. Donatella MAZZOTTA]

*Visto l'avviso del 08/07/2015 presentato dalla Sig.ra [ARMAND HUGON]***PER**

*Il Responsabile del procedimento è il Geom. [Loris Pascucci]***PER**

*F.to [Stutera Dott.ssa Giovanna]***PER**

Organization (ORG): rappresentando le ORG un mix molto eterogeneo, anche le rispettive forme nelle quali si presentano sono diverse e numerose. Nel caso di ditte o aziende l'entità è spesso preceduta da parole come "*ditta*"*, "società"*, o seguita dalle sigle *"spa"*, *"snc"*, *"srl"*, *"sas"*, ecc. Gli enti e le organizzazioni possono inoltre

presentare o il nome per esteso o la sigla puntata; nel caso in cui nello stesso periodo compaiano entrambi, questi vengono annotati separatamente.

*Il Presidente invita la [Giunta Comunale]***ORG** *ad esaminare la trattazione dell'oggetto*

*Viene così attuato il piano stabilito dal [MEF]***ORG**

*Dovranno essere presi accordi con il gestore del Servizio [UMBRIA ACQUE]***ORG**

*La centrale degli acquisti [Consip s.p.a.]***ORG** *opera nell'interesse dello Stato*

*Sentita la ditta [Boerio Candido]***ORG** *di Orio Canavese*

*Per qualsiasi segnalazione rivolgersi all'Ufficio di [Polizia Municipale]***ORG**

*Vista la delibera dell'[Autorità per la vigilanza sui contratti pubblici]***ORG** *([A.V.C.P.]***ORG***)*

Location (LOC): per quanto riguarda i luoghi non si evidenzia un'ampia variabilità formale. Le entità vengono annotate o in blocco unico nel caso di indirizzi veri e propri o altrimenti singolarmente. Tra le abbreviazioni tipicamente utilizzate, ricordiamo *"loc."* che sta per *"località"*. I nomi delle province talvolta vengono riportati per esteso, talvolta ne viene riportata solo la sigla.

*L'immobile è sito a [Genova, in Via A. Robino, n. 291R]***LOC**

*[Comune di Roma]***LOC**, *[Provincia di Roma]***LOC** *- deliberazione di Giunta*

*Lo spettacolo avrà luogo in [Abbiategrasso (MI)]***LOC**

*L'appalto viene affidato alla ditta sopra citata con sede in [loc. San Giovanni]***LOC**

*Vista la richiesta della [Comunità Montana del Pinerolese]***LOC**

*È stato proposto il programma in partenariato con la Cooperativa Alba di [Legnano]***LOC**

*Convocati presso [Comune di Mesero, via San Bernardo 41, Provincia di Mesero MI]***LOC**

Law (LAW): una certa variabilità per quanto riguarda la forma la si riscontra anche nel caso delle leggi. Una legge talvolta può essere espressa solo attraverso la dicitura *"decreto legislativo"* (o *"d.lgs",* o ancora *"decreto legge"*, o *"d.l."*, o semplicemente *"legge"*) seguito dal numero e talvolta completo anche di articolo e comma (meno frequentemente anche numero dell'allegato). Altre volte si preferisce invece il nome vero e proprio della legge (*"T.U.E.L", "Codice civile", "Codice della strada", "Codice dei contratti pubblici",* ecc. (e/o rispettive abbreviazioni)). Nei casi in cui, per una determinata legge vengano citate entrambe le forme, si valuta se è il caso di annotarle separatamente o insieme.

*Così come stabilito dalla [Legge di conversione 213/2012]***LAW**

*Visto il [D. l. 24/04/2014, n. 66]***LAW**

*Ciò viene riportato all'[art.2, comma 9 del T.U.E.L]***LAW** *emanato con [d.lgs 267/00]***LAW**

*Considerato l'[art.1 del T.U.E.L. 267/00]***LAW**

*Riportando il [Decreto del Ministero delle Infrastrutture e dei Trasporti 07/15]***LAW**

*Preso atto dell'[art. 3, allegato 11, decreto legislativo 3 marzo 2011, n. 28]***LAW**

*Ai sensi dell'[art. 892 del c.c.]***LAW** *si dichiara approvato il piano*

Act (ACT): così come le leggi, anche gli atti possono presentarsi attraverso definizioni diverse, le più frequenti *"decreto", "delibera", "determina", "determinazione", "atto"***.** Tali definizioni si presentano spesso in forma abbreviata: *"D.C.C"* sta per "Delibera di Consiglio Comunale", *"D.G.C."* per "Delibera di Giunta Comunale", queste le più frequenti. Per l'individuazione di tutte le possibili varianti sotto forma delle quali può essere espresso un atto amministrativo, abbiamo a disposizione un

documento “Atti normalizzati”. Ogni atto è caratterizzato inoltre da tipo, numero e data (solo numero e data sono obbligatori affinché un atto possa essere annotato come tale, oltre al fatto che si tratti esclusivamente di un atto proveniente da un’autorità comunale).

*Visto il [Decreto del Sindaco]***ACT_T** *[n.]***ACT_X** *[2]***ACT_N** *[del]***ACT_X** *[21.1.16]***ACT_D**

*Il piano è stato approvato con [D.G.C.]***ACT_T** *[204/15]***ACT_U**

*Richiamata la [Determinazione del Responsabile del Settore]***ACT_T** *[n.]***ACT_X** *[56]***ACT_N** *[del]***ACT_X** *[14.12.2015]***ACT_D**

*Approvato con le seguenti deliberazioni del Consiglio Comunale: [18/15]***ACT_U** *e [41/15]***ACT_U**

*[Atto di Determinazione]***ACT_T** *[n.121]***ACT_U** *[del]***ACT_X** *[03-12-2015]***ACT_D**

*Il Sindaco con [provvedimento]***ACT_T** *[del]***ACT_X** *[21-11-14]***ACT_D** *[n.]***ACT_X** *[2]***ACT_N**

*Rivista la [Determinazione C.C.]***ACT_T** *[12/09]***ACT_U**

Organization PA (PA): più omogenee dal punto di vista dell’aspetto formale, sono le entità appartenenti alla categoria delle partizioni organizzative in cui è articolato il Comune (*“Servizio Demografico”, “Ufficio Segreteria”, “Area Anagrafe”, “Tesoreria Comunale”,* ecc.). Talvolta si riscontrano anche abbreviazioni o sigle: *“Tec.”* per “tecnico”, *“U.T.C.”* per “Ufficio Tecnico”.

*Il progetto è stato approvato con deliberazione dell’[Ufficio Cultura e Turismo]***ORG_PA**

*Per ogni richiesta rivolgersi ai [Servizi Sociali]***ORG_PA** *competenti*

*F.to il Responsabile della [Direzione U.T.C.]***ORG_PA**

*Approvato con delibera dell’[Organismo Indipendente di Valutazione]***ORG_PA**

*Approvato con delibera dell'[OIV]***ORG_PA**

*Questo è quanto stabilito dal Responsabile dell'[Area Amministrativa]***ORG_PA**

*[Settore: Area Affari Generali]***ORG_PA** *[Servizio: Servizi Demografici]***ORG_PA**

5. Analisi degli errori

Per l'analisi degli errori sono stati presi in considerazione gli output ottenuti dall'applicazione dell'algoritmo di apprendimento automatico (NER), precedentemente addestrato sul training corpus, sui documenti del test corpus, messi a confronto con i risultati derivanti dall'annotazione manuale degli stessi documenti (gold standard) (vedi tab.1).

COMUNE	B-LOC	I-LOC
DI	I-LOC	I-LOC
GESSATE	I-LOC	I-LOC
Città	I-LOC	I-LOC
Metropolitana	I-LOC	I-LOC
di	I-LOC	I-LOC
Milano	I-LOC	I-LOC
Piazza	I-LOC	I-LOC
Municipio	I-LOC	I-LOC
,	I-LOC	I-LOC
n.1	I-LOC	I-LOC
20060	I-LOC	I-LOC
Gessate	I-LOC	I-LOC
(	I-LOC	I-LOC
MI	I-LOC	I-LOC
)	O	O

Tabella 1. Estratto dell'output utilizzato per l'analisi degli errori, dove la prima colonna indica il token, la seconda l'entità annotata manualmente, la terza l'entità annotata dal NER.

Dall'analisi degli errori, su un totale di 1092 entità contenute nel corpus, senza considerare le entità erroneamente non annotate (né manualmente né dal NER), si è ottenuto un tasso di errore del NER del 12,27% (134 entità).

Gli errori commessi dal NER possono essere collocati all'interno di 5 diverse classi, ognuna delle quali contraddistingue una diversa *tipologia* di errore. Di seguito si riportano le differenti classi, e qualche esempio esplicativo per ognuna.

1. Entità non annotate

[PER] *"Quinto Laura"*,
"Isolabella"

[ORG] *"Croce Rossa"*,
"SINTEL"

[LOC] *"frazione di Ferriere"*,
"regione Liguria"

[LAW] *"T.U. delle leggi sull'ordinamento degli enti locali"*,
"artt. 93, 94 del codice civile"

[ACT] *"Decreto del Sindaco del Comune di Levanto n.8 del 08/08/2014"*,
"deliberazione G.C. n. 1/2016"

[ORG_PA] *"Ufficio Tecnico Comunale"*
"Direzione lavori e contabilità dei lavori"

2. Espressioni annotate da non annotare

[PER] *"contatori"*

[ORG] *"Civica Amministrazione"*

[LOC] *"piazza di Ferriere"* (da annotare solo *"Ferriere"*)

[LAW] *"."* nell'espressione *"T.U.E.L ."* (vedi es. 1)

[ORG_PA] *"servizi comunali"*

dell'	O	O
art.13	B-LAW	B-LAW
,	I-LAW	I-LAW
comma	I-LAW	I-LAW
4	I-LAW	I-LAW
,	I-LAW	I-LAW
del	I-LAW	I-LAW
T.U.E.L	I-LAW	I-LAW
.	O	I-LAW
Lì	O	I-LAW
,	O	I-LAW

Esempio 1. Estratto dell'output utilizzato per l'analisi degli errori.

3. Entità annotate in maniera incompleta (si evidenzia la porzione non annotata)

   [PER] *"[De] Geronimi Maria"*

   [ORG] *"Ministero dell'ambiente e della tutela del territorio [e del mare]"*

   [LOC] *"Monterosso [al Mare]"*

   [LAW] *"[articolo 2, comma 2, del] decreto del Presidente della Repubblica n. 159/1999"*

   [ORG_PA] *"Settore lavori pubblici, ambiente e territorio"*

4. Entità annotate in maniera scorretta (si riporta a sinistra la corretta classe semantica)

   [PER] *"Isolabella"* (annotato come LOC)

   [ORG] *"Giacomo Rainoldi"* (annotato come PER, nell'espressione (ORG) *"Cooperativa sociale a.r.l. Giacomo Rainoldi"*

   [LOC] *"Olgiate Olona"* (annotato come PER)

   [ORG_PA] *"Ufficio polizia locale"* (nell' espressione *"Ufficio polizia locale n. 12 del 15-04-2016"* annotata come ACT)

5. Entità annotate secondo uno schema di annotazione non corrispondente a quello utilizzato

   [LOC] *"Comune di Varano Borghi, Provincia di Varese, via S. Francesco 1 – 21020 Varano Borghi"* (da annotare entro un'unica LOC) (vedi es. 2)

COMUNE	B-LOC	B-LOC
DI	I-LOC	I-LOC
VARANO	I-LOC	I-LOC
BORGHI	I-LOC	I-LOC
PROVINCIA	I-LOC	B-LOC
DI	I-LOC	I-LOC
VARESE	I-LOC	I-LOC
Via	I-LOC	I-LOC
S.	I-LOC	I-LOC
Francesco	I-LOC	I-LOC

Esempio 2. Estratto dell'output utilizzato per l'analisi degli errori.

Un ultimo caso individuato, non considerabile un vero e proprio errore, quanto piuttosto una sorta di "ambiguità" relativa allo schema di annotazione utilizzato, è costituito dall'espressione *"cooperativa "La Finestra""*, che durante l'annotazione manuale è stata annotata interamente; dal NER è stata annotata solo la denominazione dell'organizzazione in questione (vedi es. 3).

Cooperativa	B-ORG	O
"	I-ORG	O
La	I-ORG	B-ORG
Finestra	I-ORG	I-ORG
"	O	O

Esempio 3. Estratto dell'output utilizzato per l'analisi degli errori.

6. Analisi computazionali

L'analisi computazionale del corpus ha coinvolto due aspetti: quello statistico e il secondo, relativo all'accuratezza di classificazione automatica. In entrambi i casi si è proceduto con l'implementare funzioni *Python* per il calcolo dei valori ricercati.

6.1 Indici statistici

L'analisi statistica è stata effettuata sul corpus originario (il dataset costituito di 460 documenti), per ottenere informazioni relative alla grandezza del corpus, ed alla quantità di parole contenuto e di entità nominate.

I risultati ottenuti sono stati i seguenti:

- Lunghezza del corpus = 724623 tokens
- Numero totale delle *content words* presenti nel corpus = 365280 content words
- Numero totale delle entità nominate presenti nel corpus = 21329 entità
    - PER → 3705 entità
    - ORG → 3595 entità
    - LOC → 4500 entità
    - LAW → 5215 entità
    - ACT → 2238 entità
    - ORG_PA → 2076 entità
- Numero medio dei tokens per testo = 1571.850 tokens
- Numero medio delle entità nominate per testo = 46.267 entità

6.2 Indici di accuratezza di classificazione automatica

Sul test set di 25 documenti, è stato eseguito il calcolo degli indici di accuratezza statistica, attraverso la costruzione della matrice di confusione.

Per la verifica dell'affidabilità e dell'efficacia del modello è stato preso in considerazione il training corpus originario, sul quale è stata applicata la tecnica della cross-validation.

7. Matrice di Confusione e N-Fold Cross-Validation

7.1 Matrice di Confusione

Nell'ambito dell'IA una rappresentazione dell'accuratezza di classificazione statistica è data dalla *Matrice di Confusione* (detta altrimenti *Tabella di errata classificazione*).

Nella matrice di confusione ogni colonna rappresenta i *valori predetti* e ogni riga rappresenta i *valori reali* (vedi tab. 2).

	A	B	C
A	60	14	13
B	15	34	11
C	11	0	42

Tabella 2. Matrice di confusione per le classi A, B, C.

La matrice di confusione viene dunque letta in questo modo: l'elemento sulla riga i e sulla colonna j rappresenta il numero di casi in cui il classificatore ha classificato come classe i, la classe "vera" j. Questo significa che solo sulla diagonale stanno i casi classificati correttamente; gli altri sono errori. Dalla matrice di confusione sopra illustrata come esempio, costruita sulle tre classi A, B e C risulta che:

- classe A: ci sono in totale 87 casi e di questi, 60 sono stati classificati correttamente e 27 erroneamente, 14 dei quali classificati come B e 13 classificati come C.
- classe B: ci sono in totale 60 casi e di questi, 34 sono stati classificati correttamente e 26 erroneamente, 15 dei quali classificati come A e 11 classificati come C.
- classe C: ci sono in totale 53 casi e di questi, 42 sono stati classificati correttamente e 11 erroneamente come A.

La performance di un modello è determinata dal numero di predizioni corrette o, per contro, dal numero di errori di predizione. In quest'ottica, la matrice di confusione rappresenta una prima metrica per la valutazione del modello. L'utilizzo della matrice di confusione permette in particolare di osservare se vi è *confusione* nella classificazione di classi, quindi di ordinare tutti i casi del modello in categorie, determinando se il valore stimato corrisponde di volta in volta a quello definitivo.

In altre parole si tratta di un importante strumento per valutare i risultati di una stima, in quanto facilita la comprensione quindi la spiegazione degli effetti delle stime errate. Visualizzando la quantità e le percentuali in ogni cella di questa matrice, è possibile vedere con quale frequenza vengano eseguite stime accurate da parte del modello.

Nell'ambito del ML, questa tabella viene talvolta utilizzata tenendo conto delle seguenti categorie: *vero positivo*, *falso positivo*, *falso negativo* e *vero negativo* (vedi tab. 3).

Le istanze positive e negative stimate correttamente da un classificatore si definiscono rispettivamente valori veri positivi (TP) e veri negativi (TN). Analogamente, le istanze classificate in modo errato si definiscono valori falsi positivi (FP) e falsi negativi (FN). La matrice di confusione è una tabella che mostra il numero di istanze all'interno di ognuna di queste 4 categorie.

	A (PREDETTO)	**B** (PREDETTO)
A (REALE)	VERO POSITIVO	FALSO NEGATIVO
B (REALE)	FALSO POSITIVO	VERO NEGATIVO

Tabella 3. Matrice di confusione per le classi A, B.

Sulla diagonale stanno i casi classificati correttamente, dunque per la classe A in corrispondenza del valore vero positivo e per la classe B in corrispondenza del valore vero negativo. Le parole etichettate correttamente come appartenenti alla classe vengono definite *veri positivi.*

Il falso positivo, in statistica, indica il risultato di un test che erroneamente porta ad accettare l'ipotesi sulla quale esso è stato condotto; le parole etichettate erroneamente come appartenenti alla classe vengono definite *falsi positivi.*

Il falso negativo, in statistica, indica il risultato di un test che porta erroneamente a rifiutare l'ipotesi sulla quale esso è stato condotto. Le parole che non sono state etichettate come appartenenti alla classe, ma avrebbero dovuto esserlo, prendono il nome di *falsi negativi.*

Le stesse nozioni possono essere spiegate attraverso la seguente *tabella di contingenza*[23](Manning, 2008):

	Relevant	**Nonrelevant**
Retrieved	True positives	False positives
Not retrieved	False negatives	True negatives

Tabella 4. Tabella di contingenza per le classi TP, FN, FP, TN.

La matrice di confusione consente dunque di visualizzare il numero dei valori veri positivi, falsi negativi, falsi positivi e veri negativi e di ricavare inoltre alcune misure di performance:

- *Accuratezza (Accuracy)*: si definisce accuratezza la percentuale delle istanze classificate correttamente, o in altri termini, il rapporto delle classificazioni corrette del classificatore:

$$Accuracy = \frac{(tp + tn)}{(tp + fp + fn + tn)}$$

[23] Una *tabella di contingenza* è un particolare tipo di tabella comunemente utilizzata in statistica per rappresentare ed analizzare le relazioni tra due o più variabili.

È questa in genere la prima metrica che viene osservata quando si valuta un classificatore. In alcuni casi tuttavia, l'accuratezza non mostra realmente l'efficacia di un classificatore (si pensi un utente che è più interessato alle prestazioni di una classe, o il caso in cui la maggior parte delle istanze appartiene ad una delle classi). Per questo motivo risulta utile calcolare alcune metriche aggiuntive che raccolgano aspetti più specifici della valutazione.

- *Precisione (Precision)*: la precisione rappresenta la percentuale dei valori positivi classificati correttamente tra quelli identificati come appartenenti alla classe:

$$Precision = \frac{\#(relevant\ items\ retrived)}{\#(retrieved\ items)} \quad \text{o} \quad Precision = \frac{TP}{TP + FP}$$

- *Richiamo (Recall)*: il richiamo (o *tasso vero positivo*) rappresenta la percentuale dei valori positivi classificati correttamente tra quelli che effettivamente appartengono alla classe. È dunque il risultato del rapporto tra i veri positivi e la somma di veri positivi e falsi negativi:

$$Recall = \frac{\#(relevant\ items\ retrieved)}{\#(relevant\ items)} \quad \text{o} \quad Recall = \frac{TP}{TP + FN}$$

- *F-score*: si tratta di un'altra metrica spesso utilizzata, che prende in considerazione precisione e richiamo. Si tratta della media armonica[24] delle due metriche ed è calcolata nel modo seguente:

[24] La media armonica di n numeri, in aritmetica, è definita come il reciproco della media aritmetica dei reciproci.

$$F - score = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)}{\beta^2 P + R} \quad \text{dove} \quad \beta^2 = \frac{1-\alpha}{\alpha}$$

Dove α ∈ [0,1] quindi $\beta^2 \in [0, \infty]$ e $\alpha = \frac{1}{2}$ o $\beta = 1$.

Questa viene comunemente scritta come F_1 che è l'abbreviazione di $F_{\beta=1}$. Quando si assume $\beta = 1$ la formula si semplifica in:

$$F_{\beta=1} = \frac{2PR}{P+R}.$$

Attraverso l'applicazione della matrice di confusione al test set, si sono ottenuti, per ogni singola classe, i valori di veri positivi, falsi negativi e falsi positivi, rappresentati nel grafico che segue.

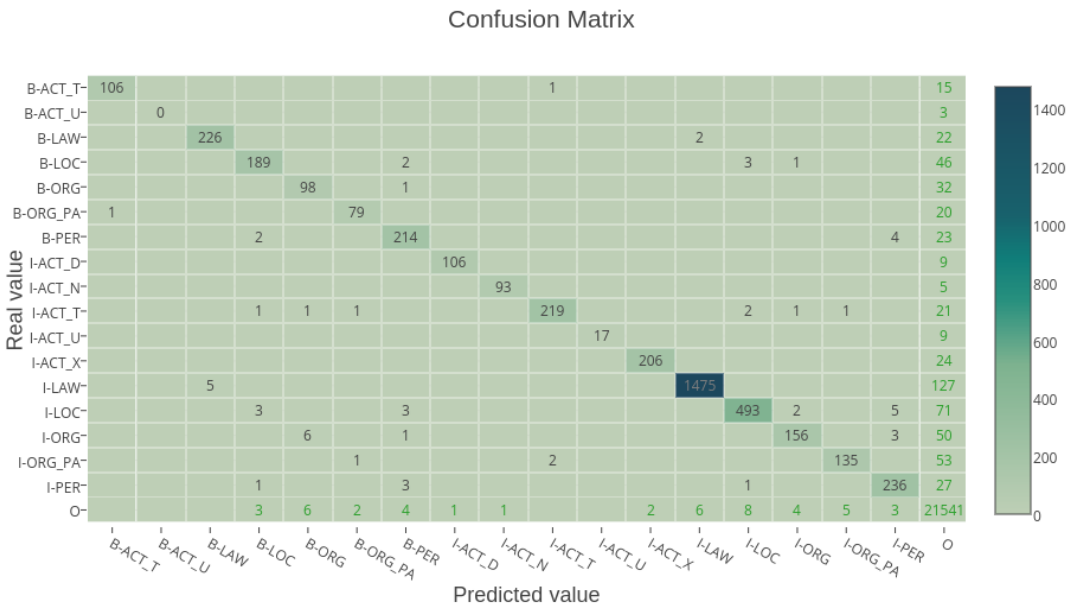

Grafico 1 Matrice di confusione calcolata sul test set.

Dall'analisi del grafico, risulta che:

- B-PER: 214 casi correttamente annotati, 11 casi erroneamente annotati, 27 casi erroneamente non annotati (4 di questi annotati come I-PER);

- I-PER: 236 casi correttamente annotati, 3 casi erroneamente annotati, 27 casi erroneamente non annotati;
- B-ORG: 98 casi correttamente annotati, 13 casi erroneamente annotati, 33 casi erroneamente non annotati (1 di questi annotato come B-PER);
- I-ORG: 156 casi correttamente annotati, 4 casi erroneamente annotati, 53 casi erroneamente non annotati (3 di questi annotati come I-PER);
- B-LOC: 189 casi correttamente annotati, 10 casi erroneamente annotati, 52 casi erroneamente non annotati (2 di questi annotati come B-PER, 3 come I-LOC, 1 come I-ORG);
- I-LOC: 493 casi correttamente annotati, 9 casi erroneamente annotati, 78 casi erroneamente non annotati (2 di questi annotati come I-ORG, 5 come I-PER);
- B-LAW: 226 casi correttamente annotati, 5 casi erroneamente annotati, 24 casi erroneamente non annotati (2 di questi annotati come I-LAW);
- I-LAW: 1475 casi correttamente annotati, 6 casi erroneamente annotati, 127 casi erroneamente non annotati;
- B-ACT_T: 106 casi correttamente annotati, 1 caso erroneamente annotato, 16 casi erroneamente non annotati (1 di questi annotato come I-ACT_T);
- I-ACT_T: 219 casi correttamente annotati, 2 casi erroneamente annotati, 25 casi erroneamente non annotati (2 di questi annotati come I-LOC, 1 come I-ORG, 1 come I-ORG_PA);
- I-ACT_X: 206 casi correttamente annotati, 2 casi erroneamente annotati, 24 casi erroneamente non annotati;
- I-ACT_N: 93 casi correttamente annotati, 1 caso erroneamente annotato, 5 casi erroneamente non annotati;
- I-ACT_D: 106 casi correttamente annotati, 1 caso erroneamente annotato, 9 casi erroneamente non annotati;
- B-ACT_U: 0 casi correttamente annotati, 0 casi erroneamente annotati, 3 casi erroneamente non annotati;
- I-ACT_U: 17 casi correttamente annotati, 0 casi erroneamente annotati, 9 casi erroneamente non annotati;
- B-ORG_PA: 79 casi correttamente annotati, 4 casi erroneamente annotati, 20 casi erroneamente non annotati;

- I-ORG_PA: 135 casi correttamente annotati, 5 casi erroneamente annotati, 53 casi erroneamente non annotati.

7.2 N-Fold Cross-Validation

Per la verifica dell'accuratezza del campione (o modello) è stata applicata, al training set la tecnica della *N-Fold Cross-Validation*.

La n-fold cross-validation (o cross-validation) è una tecnica statistica, applicabile in presenza di una buona numerosità del campione analizzato (il training set). La n-fold cross-validation consiste nella suddivisione del corpus (il dataset totale) in sottoinsiemi di uguale numerosità (n), per ognuno dei quali, ad ogni passo, l'ennesima parte del dataset viene a costituire il test e la restante il training. Se ad ogni step, per ogni pacchetto viene utilizzato il 10% dei documenti come test e il 90% come training, alla fine si avrà che di tutti i documenti costituenti il corpus, ognuno è stato utilizzato almeno una volta come set ed almeno una volta come training. In altre parole, si suddivide il campione in gruppi di egual numerosità, si esclude iterativamente un gruppo (test) alla volta e lo si cerca di predire attraverso i gruppi non esclusi (che costituiscono il training). In questo modo si allena il modello, evitando problemi di *overfitting*[25], ma anche di *campionamento asimmetrico* del training dataset, tipico della suddivisione del dataset in due sole parti (training e test).

La cross-validation costituisce una tecnica standard utilizzata nell'ambito del ML, che permette di valutare la variabilità del set di dati sia l'affidabilità di qualsiasi modello sottoposto a training utilizzando tali dati.

Attraverso l'elaborazione di un foglio di calcolo Excel, è stato calcolato inoltre il valore della *macro-average*.

Nel caso dell'elaborazione di una raccolta con classificatori a più classi, risulta spesso utile calcolare un'unica misura aggregata che combina le misure di precisione,

[25] Nel campo del ML, si parla di *overfitting* quando, per talune circostanze, come uno scarso numero di esempi di training, accade che il modello si adatta a caratteristiche che sono specifiche solo del training set e che spesso non hanno riscontro nel resto dei casi. In presenza di overfitting, le prestazioni sui dati di training aumentano, a differenza delle prestazioni sui dati non visionati, che saranno peggiori.

richiamo e f_1-score, ottenute per ogni singola classe. Il calcolo della media è possibile attraverso due differenti metodi denominati rispettivamente: *macroaveraging* e *microaveraging*.

Il primo calcola una media semplice su classi, in altri termini, media tutte le decisioni prese globalmente (TP, FP, FN, TN) calcolando i singoli indici sull'intero spazio classi-testi. È questo il metodo che più si adatta a sistemi di Information Retrieval.

La macroaverage può esser definita anche come la media dei singoli valori di precisione e di richiamo ottenuti a seguito di n-interrogazioni.

Il secondo raggruppa per testo le decisioni tra le classi, poi calcola una misura di efficacia sulla tabella di contingenza aggregata; in altre parole, media i parametri calcolati ottenuti per ogni singola classe a posteriori.

La differenza tra i due metodi può essere grande. Sostanzialmente macroaveraging dà un ugual peso a tutte le classi, mentre microaveraging dà un ugual peso a tutte le decisioni di classificazione per documento, a tutti i testi. Poiché la metrica F_1 ignora i veri negativi e il suo valore è determinato per la maggior parte dal numero dei veri positivi, le classi più ampie dominano le più piccole, nell'ambito del microaveraging.

I valori ottenuti sono stati i seguenti:

Entity	Precision	Recall	F_1-score
ACT	0,79284	0,86971	0,82647
LAW	0,82675	0,83815	0,83236
LOC	0,70917	0,74939	0,72746
ORG	0,70853	0,68194	0,69407
PER	0,83366	0,86593	0,84895
ORG_PA	0,60898	0,77885	0,6819
Macro AVG	**0,76274**	**0,8166**	**0,78834**

Tabella 5. Risultati dello Stanford NER addestrato sul training corpus.

Dall'analisi della tabella risulta che:

- Per le classi PER, LOC, LAW, ACT, ORG_PA:

  $$precisione < richiamo$$

  ciò significa che per queste classi, il numero di falsi positivi è maggiore rispetto al numero di falsi negativi. In altre parole è maggiore il numero dei casi, per ognuna delle classi, erroneamente annotati come appartenenti alla classe, rispetto al numero dei casi, per ognuna delle classi, erroneamente non annotati, ma che avrebbero dovuto esserlo.
- Per la classe ORG:

  $$precisione > richiamo$$

  ciò significa che per questa classe, il numero di falsi positivi è minore rispetto al numero di falsi negativi. In altre parole è maggiore il numero dei casi, per ognuna delle classi, erroneamente non annotati, ma che avrebbero dovuto esserlo, rispetto al numero dei casi, per ognuna delle classi, erroneamente annotati come appartenenti alla classe.
- La media ottenuta mediante l'aggregazione delle classi, per i valori di *precisione* risulta pari a 0,76274.
- La media ottenuta mediante l'aggregazione delle classi, per i valori di *richiamo* risulta pari a 0,8166.
- La media ottenuta mediante l'aggregazione delle classi, per i valori di f_1-*score* risulta pari a 0,78834.

8. Conclusioni

Il presente studio è nato con lo scopo, dopo aver preso in esame la variabilità delle entità nominate (PER, ORG, LOC, LAW, ACT, ACT_PA, nel nostro caso), di analizzare gli errori commessi dal NER adattato al dominio della Pubblica Amministrazione.

Dapprima è stata annotata manualmente la classe dei nomi presenti all'interno della collezione di testi amministrativi (che ha costituito il training set), con lo scopo di perfezionare le prestazioni dell'algoritmo di apprendimento automatico. Dopo si è passati all'addestramento (sul training set), nonché al test del NER su un nuovo corpus, il test.

Allo studio della variabilità delle entità nominate ed all'analisi degli errori è seguita l'analisi degli indici di accuratezza di classificazione statistica, relativi al training set ed al test. Al training set è stata applicata la tecnica della cross-validation, al fine di valutare l'efficacia del modello. Attraverso il calcolo della matrice di confusione sul test, si è ottenuta una stima circa l'accuratezza di classificazione del NER.

Per quanto riguarda gli sviluppi sul NER, sono in corso gli studi per il miglioramento delle performances, da un lato orientati verso l'ottimizzazione delle features per l'addestramento, dall'altro alla revisione ed estensione delle regole per ampliare il processo di estrazione includendo anche le classi *più deboli* (nel nostro caso per es. ORG_PA).

9. Bibliografia

Andrieu, Christophe, Arnaud Doucet, Nando De Freitas, Michael I. Jordan. 2003. *An introduction to MCMC for Machine Learning*. "Machine Learning", 50, pp. 5-43.

Appelt, Douglas E. 1999. *Introduction to Information Extraction.* "AI Communications", 12, pp. 161-172.

Bontcheva, Kalina, Hamish Cunningham, Diana Maynard, Horacio Saggion, Valentin Tablan, Cristian Ursu, Yorick Wilks. 2002. *Architectural Elements of Language Engineering Robustness*. "Natural Language Engineering", 8, pp. 257-274.

Borthwick, Alan. 1999. *A Maximum Entropy Approach to Named Entity Recognition*. Ph.D. thesis, New York University.

Bunescu, Razvan, Raymond J. Mooney. 2004. *Collective Information Extraction with Relational Markov Networks*. "Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics", 2004, pp. 439-446.

Chieu, Hai Leong, Hwee Tou Ng. 2002. *Named Entity Recognition with a Maximum Entropy Approach*. "Proceedings of the 7th Conference on Natural Language Learning", 4, pp. 160-163.

Cunningham, Hamish. 1997. *Information Extraction: a User Guide*. Sheffield, University of Sheffield.

Curran, James R., Clark Stephen. 2003. *Language Independent NER using a Maximum Entropy Tagger*. "Proceedings of the 7th Conference on Natural Language Learning", 4, pp. 164-167.

Ferreira da Silva, Joaquim F., Zornitsa Kozareva, José Gabriel Pereira Lopes. 2004. *Cluster Analysis and Classification of Named Entities*. "Proceedings of the Conference on Language Resources and Evaluation", 2004.

Finkel Jenny Rose, Trond Grenager, Christopher Manning. 2005. *Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling*. "Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics", 2005, pp. 363-370.

Freitag, Dayne, Andrew Kachites McCallum. 1999. *Information Extraction with HMMs and Shrinkage*. "Proceedings of the Workshop on Machine Learning for Information Extraction", 1999.

Jansche, Martin. 2002. *Named Entity Extraction with Conditional Markov Models and Classifiers.* "Proceeding of the 6th Conference on Natural Language Learning", 20, pp. 1-4.

Jurafsky, Daniel, James H. Martin. 2009. *Speech and Language Processing: An introduction to natural language processing, computational linguistics and speech recognition.* Upper Saddle River, N.J., Prentice Hall.

Lafferty, John, Andrew McCallum, Fernando Pereira. 2001. *Conditional Random Fields: Probabilistics models for segmenting and labeling sequence data*. "Proceedings of the 18th International Conference on Machine Learning", 2001, pp. 282-289.

Leek, Timothy R., 1997. *Information Extraction using hidden Markov models.* Master's thesis, U.C. San Diego.

Magnini et al. 2011. *ITALIAN CONTENT ANNOTATION BANK (I-CAB): Named Entities*.

Magnini, et al. 2006. *Annotazione di contenuti concettuali in un corpus italiano: I-CAB*. "Prospettive nello studio del lessico italiano: atti del IX Congresso della Società Internazionale di Linguistica e Filologia Italiana", 1, pp. 321-328.

Manning, Christopher D., Prabhakar Raghavan, Hinrich Schütze. 2008. *An Introduction to Information Retrieval.* Cambridge, Cambridge University Press.

McCallum, Andrew, Wei Li. 2003. *Early Results for Named Entity Recognition with Conditional Random Fields, Features Induction and Web-Enhanced Lexicons.* "Proceedings of the 7th Conference on Natural Language Learning", 4, pp. 188-191.

Melucci, Massimo. 2013. *Information Retrieval: Metodi e modelli per i motori di ricerca*. Padova, Franco Angeli.

Nadeau, David, Satoshi Sekine. 2007. *A Survey of Named Entity Recognition and Classification.* "Linguisticae Investicationes", 30, pp. 3-26.

Passaro, Lucia C., Alessandro Lenci. 2014. *Deliverable 6: Specifiche degli Strumenti per l'annotazione semantica*. "SEMantic instruments for PubLIc administrators and CitizEns", 2014.

Rabiner, Lawrance R. 1989. *A Tutorial on Hidden Markov Models and selected applications in Speech Recognition*. "Proceedings of the IEEE", 77, pp. 257-286.

Rau, Lisa F. 1991. *Extracting Company Names from Text*. "Proceedings of the Conference on Artificial Intelligence Application", 1991.

Riloff, Ellen M. 1994. *Information Extraction as a Basis for Portable Text Classification System.* Amherst, Università del Massachussets.

Sebastiani, Fabrizio. 2002. *Machine learning in automated text categorization*. "ACM Computing Surveys", 34, pp. 1-47.

Tjong Kim Sang, Erik F. 2002. *Introduction to the CoNLL-2002 Shared-Task: Language-Indipendent Named Entity Recognition*. "Proceedings of the Conference on Natural Language Learning", 2003.

Tjong Kim Sang, Erik F., Fien De Meulder. 2003. *Introduction to the CoNLL-2002 Shared-Task: Language-Indipendent Named Entity Recognition*. "Proceedings of the 7th Conference on Natural Language Learning", 4, pp. 142-147.

Van Asch, Vincent. 2012. *Macro- and micro-averaged evaluation measures*. Antwerp, Università di Antwerp.

Van Rijsbergen, Cornelis J. 1979. *Information Retrieval*. London, Butterworths.

Viterbi, Andrew J. 1967. *Error bounds for convolutional codes and an asymptotically optimum decoding algorithm*. "IEEE Transactions on Information Theory", 13, pp. 260-269.

Yang, Yiming, Xin Liu. 1999. *A re-examination of text categorization methods*. "Proceedings of the 22nd Annual International Conference on Research and Development in Information Retrieval", 1999, pp. 42-49.

9.1 Sitografia

CELI: Language Technology, *Che cos'è il Natural Language Processing*

https://www.celi.it/blog/2015/10/che-cose-il-natural-language-processing/

Cs.uregina, *Confusion Matrix*

http://www2.cs.uregina.ca/~dbd/cs831/notes/confusion_matrix/confusion_matrix.html

Gallerani, *Natural Language Processing (NLP) e Information Extraction*

http://www.gallerani.it/sito/natural-language-processing-nlp-e-information-extraction-ie/

Microsoft Azure, *Come valutare le prestazioni del modello in Azure Machine Learning*

https://azure.microsoft.com/it-it/documentation/articles/machine-learning-evaluate-model-performance/

Wikipedia, voce *Conditional random field*

https://en.wikipedia.org/wiki/Conditional_random_field#Inference

Wikipedia, voce *Matrice di confusione*

https://it.wikipedia.org/wiki/Matrice_di_confusione

Wikipedia, voce *Media armonica*

https://it.wikipedia.org/wiki/Media_armonica

Wikipedia, voce *Overfitting*

https://it.wikipedia.org/wiki/Overfitting